

(19)

Europäisches Patentamt
European Patent Office
Office européen des brevets



(11)

EP 1 187 343 A2

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
13.03.2002 Bulletin 2002/11

(51) Int Cl.7: **H03M 13/41**

(21) Application number: **01307321.8**

(22) Date of filing: **29.08.2001**

(84) Designated Contracting States:
**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE TR**
Designated Extension States:
AL LT LV MK RO SI

(72) Inventor: **Kuwazoe, Yasuyoshi,**
Intell. Prop. Dept. Sony Corp.
Tokyo 141-0001 (JP)

(30) Priority: **31.08.2000 JP 2000264036**

(74) Representative: **Pilch, Adam John Michael et al**
D. YOUNG & CO.,
21 New Fetter Lane
London EC4A 1DA (GB)

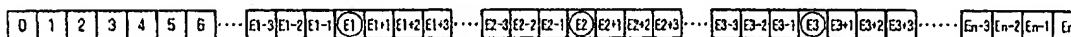
(71) Applicant: **Sony Corporation**
Tokyo 141-0001 (JP)

(54) Data demodulation apparatus and methods

(57) A data decoding technique in which the data rate can be discriminated at a high speed uses a Viterbi decoding process. Receive data are successively Viterbi decoded beginning with the top bit (0) thereof. At a point of time when receive data which may possibly be

end bit (E1, E2, E3) position data required for data rate discrimination are Viterbi decoded, data necessary for the data rate discrimination are successively extracted, and the data rate is discriminated based on the extracted necessary data.

FIG. 4



EP 1 187 343 A2

Description

[0001] This invention relates to a data demodulation apparatus and method, and more particularly to a data demodulation apparatus and method which uses a Viterbi decoding process to discriminate the data rate of received data.

[0002] A data rate communication method called blind transport format detection is known as one of data rate discrimination methods for a communication system in which a plurality of data rates are used.

[0003] According to the blind transport format detection, the data rate is discriminated in the following manner.

[0004] In particular, a data demodulation apparatus in a communication system in which a plurality of data rates are used cannot recognize the data rate at a point of time at which it receives data. More particularly, the number of possible data rates is prescribed in advance, and therefore, the data demodulation apparatus recognizes such possible data rates, but cannot recognize which one of the data rates is actually used by the receive data. Therefore, the data demodulation apparatus Viterbi decodes the received data and determines a maximum path metric value, a minimum path metric value and a zero-state path metric value for each data of the end bit position n_{end} which may possibly be the end of an amount of data corresponding to the data rate as counted from the top of the receive data. Then, the data demodulation apparatus calculates the S value (a value represented by a function $S(n_{\text{end}})$) given by the following expression (1):

$$S(n_{\text{end}})$$

$$= 10 \log(((a_0(n_{\text{end}}) - a_{\min}(n_{\text{end}})) / ((a_{\max}(n_{\text{end}}) - a_{\min}(n_{\text{end}}))) \quad (1)$$

where $a_{\max}(n_{\text{end}})$ is the maximum path metric value at the end bit n_{end} , $a_{\min}(n_{\text{end}})$ is the minimum path metric value at the end bit n_{end} , and $a_0(n_{\text{end}})$ is the zero-state path metric value at the end bit n_{end} .

[0005] If the possible end bit is a true end bit, then the zero-state path metric value $a_0(n_{\text{end}})$ exhibits a sufficiently low value, and therefore, the S value given by the expression (1) above has a sufficiently small value (a high value in the negative direction). On the other hand, if the possible end bit is not a true end bit, then the zero-state path metric value $a_0(n_{\text{end}})$ cannot assume a sufficiently low value, and therefore, the S value does not have a small value.

[0006] Thus, making use of the relationship just described, the S value calculated with regard to a predetermined end bit n_{end} is compared with a predetermined threshold value D. Then, if the relationship given by the following expression (2) is not satisfied, then it is discriminated that the end bit n_{end} is not a true end bit, and the S value of another end bit is determined.

$$S(n_{\text{end}}) \leq D \quad (2)$$

where the initial value of the threshold value D is set to a comparatively high value.

[0007] On the other hand, if the S value satisfies the expression (2) above, then the data demodulation apparatus executes a CRC (Cyclic Redundancy Check) check from the zero-state path memory data. If the CRC check does not reveal an error, then the data demodulation apparatus sets the S value at the end at the point of time to the threshold value D. This operation is repeated until the end bit n_{end} that has the lowest S value (that is, the data length from the top data to the end bit n_{end}) is finally detected as the data rate.

[0008] It is assumed that the data demodulation apparatus receives, for example, such receive data as seen in Fig. 1. The receive data of Fig. 1 includes a variable length data part (Data with variable number of bits) 1, a CRC part (CRC) 2, and an empty data part (Empty) 3.

[0009] If it is assumed that end bits E1 to E4 may possibly be an end bit n_{end} as seen in Fig. 1, that is, if it is assumed that four data rates from the top (left end in Fig. 1) of the receive data to the end bits E1, E2, E3 and E4 are set in advance, then the data demodulation apparatus determines the S value with regard to each of the end bits E1 to E4 and repetitively compares the S values with the threshold value as described above. In the receive data of Fig. 1, the end bit E3 forms a break of the CRC part and makes a true end bit. Accordingly, the S value exhibits a minimum value at the end bit E3, and as a result, the data length from the top to the end bit E3 of the received data is discriminated as the data rate.

[0010] In the blind transport format detection described above, the arithmetic operation of the maximum path metric value $a_{\max}(n_{\text{end}})$, minimum path metric value $a_{\min}(n_{\text{end}})$ and zero-state path metric value $a_0(n_{\text{end}})$ with regard to each possible end bit involves repetitions of a Viterbi decoding process from the top bit to all end bits which each may possibly be a true end bit.

[0011] In particular, for example, referring to Fig. 2, in order to determine the maximum path metric value $a_{\max}(E1)$,

minimum path metric value $a_{\min}(E1)$ and zero-state path metric value $a_0(E1)$ with regard to the end bit E1 in the top stage in Fig. 2, the Viterbi decoding process is performed for each of the bits from the top bit 0 to the end bit E1 in the top stage in Fig. 2.

[0012] Meanwhile, in order to determine the maximum path metric value $a_{\max}(E2)$, minimum path metric value $a_{\min}(E2)$ and zero-state path metric value $a_0(E2)$ with regard to the end bit E2 in the second top stage in Fig. 2, the Viterbi decoding process is performed for each of the bits from the top bit 0 to the end bit E2 in the second top stage in Fig. 2.

[0013] If n bit rates are set in advance as seen in Fig. 2, such a Viterbi decoding process as described above is repeated to determine the maximum path metric value $a_{\max}(En)$, minimum path metric value $a_{\min}(En)$ and zero-state path metric value $a_0(En)$ with regard to each of the end bits E1 to En, and the S values determined based on the thus determined path metric values are compared and the data length from the top bit to the end bit En corresponding to S_{\min} is discriminated as the bit rate.

[0014] However, if the Viterbi decoding process is executed in accordance with the procedure described above to perform bit rate discrimination, then since a similar Viterbi decoding process must be repeated by a number of times equal to the number of possible end bits, that is, the set number of data rates, much time is required for arithmetic operation to finally detect a true end bit. As a result, the technique described above has a subject to be solved in that much time is required for discrimination of the bit rate.

[0015] In order to overcome the difficulty just described, it is a possible idea to use a plurality of Viterbi decoders such that the above-described Viterbi decoding process is executed parallelly thereby to perform the necessary arithmetic operation at an increased speed. This, however, results in another problem in that the apparatus scale is increased and also the cost is increased.

[0016] It is an aim of at least an embodiment of the present invention to provide a data decoding apparatus and method by which the bit rate of receive data can be discriminated at a high speed by a Viterbi decoding process with a simple apparatus configuration.

[0017] According to an aspect of the present invention, there is provided a data demodulation apparatus for receiving data of an arbitrary one of a plurality of data rates and demodulating the receive data making use of blind transport format detection, comprising Viterbi decoding means for successively Viterbi decoding the receive data beginning with the top bit, end bit position discrimination data extraction means for successively extracting end bit position discrimination data which are obtained at end bit positions of the data rates when the receive data are successively Viterbi decoded beginning with the top bit by the Viterbi decoding means, and data rate discrimination means for discriminating the data rate of the receive data based on the end bit position discrimination data successively extracted by the end bit position discrimination data extraction means.

[0018] Preferably, the data demodulation apparatus further comprises end bit position discrimination data storage means for storing the end bit position discrimination data successively extracted by the end bit position discrimination data extraction means.

[0019] Preferably, the Viterbi decoding means Viterbi decodes a plurality of bits of the receive data in one step.

[0020] If a bit of the receive data which may possibly be an end bit position bit is not the last bit of the plurality of bits which are Viterbi decoded in one step by the Viterbi decoding means, then the Viterbi decoding means may first Viterbi decode those of the receive data to be Viterbi decoded in one step from the top of the receive data to the bit which may possibly be an end bit position bit, and then Viterbi decode the remaining one or ones of the receive data up to the last bit of the plurality of bits.

[0021] In this instance, the data demodulation apparatus may further comprise Viterbi-decoded data storage means for storing data Viterbi decoded by the Viterbi decoding means, and, if the bit of the receive data which may possibly be an end bit position bit is not the last bit of the plurality of bits which are Viterbi decoded in one step by the Viterbi decoding means, then when the Viterbi decoding means Viterbi decodes those of the receive data to be Viterbi decoded in one step from the top of the receive data to the bit which may possibly be an end bit position bit, the Viterbi-decoded data storage means may be inhibited from storing the Viterbi decoded receive data.

[0022] The end bit position discrimination data may include a maximum path metric value, a minimum path metric value, a zero-state path metric value and zero-state path memory data.

[0023] The present invention also provides a corresponding data demodulation method.

[0024] With the data demodulation apparatus and method, receive data are successively Viterbi decoded beginning with the top bit. Then, end bit position discrimination data which are obtained at end bit positions of data rates when the receive data are successively Viterbi decoded beginning with the top bit are successively extracted. Finally, the data rate of the receive data is discriminated based on the end bit position discrimination data successively extracted in this manner.

[0025] The invention will now be described by way of example with reference to the accompanying drawings, throughout which like parts are referred to by like references, and in which:

Fig. 1 is a diagrammatic view illustrating discrimination of a data rate;

Fig. 2 is a diagrammatic view illustrating a conventional Viterbi decoding process;

Fig. 3 is a block diagram showing a configuration of a data demodulation apparatus to which an embodiment of the present invention is applied;

Fig. 4 is a diagrammatic view illustrating receive data;

5 Fig. 5 is a flow chart illustrating a process executed by the data demodulation apparatus of Fig. 3;

Figs. 6A and 6B are diagrammatic views illustrating a receive data Viterbi decoding process by the data demodulation apparatus of Fig. 3;

Fig. 7 is a flow chart illustrating details of an EBP process by the data demodulation apparatus of Fig. 3;

Fig. 8 is a flow chart illustrating details of a CRC process by the data demodulation apparatus of Fig. 3; and

10 Fig. 9 is a flow chart illustrating details of an EBP data extraction process by the data demodulation apparatus of Fig. 3.

[0026] Referring to Fig. 3, there is shown a data demodulation apparatus to which an embodiment of the present invention is applied. The data demodulation apparatus shown is employed in a communication system in which a plurality of data rates are used, and which discriminates the data rate of receive data, demodulates the receive data at the appropriate data rate and outputs the decoded data to an apparatus of the next stage not shown. To this end, the data demodulation apparatus includes a receive data memory 11 for temporarily storing receive data inputted thereto through an antenna or a demodulator not shown in accordance with an instruction of a control section 12. The receive data memory 11 outputs the stored data for each predetermined number of bits to a receive data control section 13 in accordance with another instruction of the control section 12.

[0027] The control section 12 may include a CPU (Central Processing Unit), a ROM (Read Only Memory) and a RAM (Random Access Memory) or logic circuitry all not shown and controls operation of the entire data demodulation apparatus. The control section 12 includes an internal counter not shown, which counts receive data inputted to the receive data memory 11 in a unit of a bit and stores the count value. The control section 12 executes various processes based on the count data of the received data by the built-in counter. The control section 12 further has built therein a counter for the S value described hereinabove and a data rate counter for detecting the data rate.

[0028] The receive data control section 13 supplies receive data inputted thereto from the receive data memory 11 to an ACS (Add Compare Select) circuit 14 in accordance with an instruction of the control section 12. Further, when necessary, the receive data control section 13 performs a masking process for receive data (process for changing some of receive data to 0) and supplies resulting data to the ACS circuit 14. The masking process for receive data of the receive data control section 13 is hereinafter described in detail.

[0029] The ACS circuit 14 performs such processes as addition, comparison and selection, that is, a Viterbi decoding process, for received data inputted thereto from the receive data control section 13 using processed data stored in a path metric path memory 15 in accordance with an instruction of the control section 12, and outputs a result of the processes so as to be stored into the path metric path memory 15. Further, the ACS circuit 14 stores EBP (End Bit Position) data, that is, a maximum path metric value, a minimum path metric value, a zero-state path metric value and zero-state path memory data of a bit which may possibly be an end bit displaced by an amount corresponding to the data rate from the top bit of the receive data. It is to be noted that the ACS circuit 14 executes a Viterbi decoding process for a plurality of bits in one step.

[0030] The path metric path memory 15 stores Viterbi-decoded data inputted thereto from the ACS circuit 14 in accordance with an instruction of the control section 12. Then, after the data rate is discriminated, the path metric path memory 15 outputs the stored decoded data at the correct data rate to an apparatus of the following stage not shown. Further, the path metric path memory 15 supplies such Viterbi-decoded data as data necessary for Viterbi decoding of next receive data to the ACS circuit 14.

45 [0031] A CRC (Cyclic Redundancy Check) circuit 17 performs CRC calculation, that is, executes a CRC check, using decoded data positioned preceding to zero-state path memory data stored in an EBP memory 16 and zero-state path memory data stored in the path metric path memory 15 to discriminate whether or not the receive data include some errors. The CRC circuit 17 outputs a result of the discrimination as a flag of 0 or 1 to the control section 12.

[0032] Now, operation of the data demodulation apparatus where the ACS circuit 14 processes such receive data as illustrated in Fig. 4 three by three bits by Viterbi decoding of one step is described with reference to a flow chart of Fig. 5. It is to be noted that, in the following description, r represents the number of bits which can be Viterbi decoded at a time in one step, and accordingly, in the case just described, $r = 3$. Further, it is assumed that the counter value and the bit number of receive data are equal to each other, and, for example, the bit of receive data corresponding to the counter value M is the bit M (this may be hereinafter referred to as receive data M). Furthermore, those data of the receive data illustrated in Fig. 4 which may possibly be EBP data are each denoted by an encircling mark and represented as receive data $E1$ to En .

55 [0033] Referring to Fig. 5, in step S1, the control section 12 initializes the built-in counters to set the counter $M = 0$, $S_{min} = D$ and the data rate = 0.

[0034] In step S2, the receive data memory 11 stores receive data and outputs receive data M to M+r-1 to the receive data control section 13 in accordance with an instruction of the control section 12. In particular, in the first processing, since the counter value is $M = 0$, the receive data memory 11 outputs receive data 0 to 2 ($= 0 + 3 - 1$) illustrated in Fig. 4 to the receive data control section 13.

[0035] In step S3, the control section 12 discriminates whether or not the receive data M to M+r-1 include receive data (a bit) which may possibly be EBP data. In particular, in the present case, the control section 12 discriminates whether or not the receive data 0 to 2 include data which may possibly be EBP data. Since the receive data 0 to 2 does not include data which may possibly be EBP data as seen in Fig. 4, the processing advances to step S4.

[0036] In step S4, the receive data control section 13 outputs the receive data M to M+r-1 to the ACS circuit 14 in accordance with an instruction of the control section 12. In particular, in the present case, the receive data control section 13 outputs the receive data 0 to 2 to the ACS circuit 14.

[0037] In step S5, the ACS circuit 14 Viterbi decodes the receive data M to M+r-1 inputted thereto from the receive data control section 13 and outputs a result of the Viterbi decoding to the path metric path memory 15. Accordingly, in the present case, the ACS circuit 14 Viterbi decodes the receive data 0 to 2 and outputs a result of the Viterbi decoding to the path metric path memory 15.

[0038] In step S6, the control section 12 increments the counter value M by r, that is, by 3. Thereafter, the processing returns to step S2.

[0039] In step S2, if the counter value is, for example, $M = E1 - 2$, then the receive data memory 11 outputs the receive data E1-2 to E1 to the receive data control section 13 in accordance with an instruction of the control section 12.

[0040] In step S3, the control section 12 discriminates whether or not the receive data E1-2 to E1 inputted thereto from the receive data memory 11 include receive data which may possibly be EBP data. In the present case, since the receive data E1 may possibly be EBP data, the control section 12 discriminates that data which may possibly be EBP data is present within the range of the receive data E1-2 to E1. Thus, the processing now advances to step S7.

[0041] In step S7, the control section 12 discriminates whether or not the receive data M+r-1 is data which may possibly be EBP data. In particular, in step S7, the control section 12 discriminates, when it processes the receive data E1-2 to E1, whether or not the receive data which may possibly be EBP data is the last data (bit) to be Viterbi decoded in one step by the ACS circuit 14 as seen in Fig. 6A. In the present case, the receive data M+r-1 is the receive data E1. The receive data E1 is data which may possibly be EBP data. Accordingly, the processing advances to step S8.

[0042] It is to be noted that Fig. 6A is an enlarged view illustrating those of the receive data of Fig. 4 which are in the proximity of the top data and the receive data E1 and E2, and each range indicated by a double-sided arrow mark represents a set of receive data which are to be Viterbi decoded in one step by the ACS circuit 14 when the counter has a count value given below the double-sided arrow mark. Meanwhile, Fig. 6B illustrates processes of the ACS circuit 14 corresponding to the counter values illustrated in Fig. 6A.

[0043] In step S8, the receive data control section 13 outputs the receive data M to M+r-1 to the ACS circuit 14 in accordance with an instruction of the control section 12. In the present case, the receive data control section 13 outputs the receive data E1-2 to E1 to the ACS circuit 14. In step S9, the ACS circuit 14 makes use of data, which have been obtained by the Viterbi decoding in the past and are stored in the path metric path memory 15, to Viterbi decode the receive data E1-2 to E1 inputted thereto from the receive data control section 13 and stores a result of the Viterbi decoding into the path metric path memory 15. Further, the ACS circuit 14 stores a maximum path metric value, a minimum path metric value, a zero-state path metric value and zero-state path memory data with regard to the receive data E1 into the EBP memory 16.

[0044] In step S10, the control section 12 executes an EBP process.

[0045] The EBP process is described in more detail with reference to a flow chart of Fig. 7.

[0046] In step S21, the control section 12 reads out the maximum path metric value, minimum path metric value and zero-state path metric value of receive data, which may possibly be EBP data, from the EBP memory 16. In the present case, the control section 12 reads out the maximum path metric value, minimum path metric value and zero-state path metric value of the receive data E1 from the EBP memory 16.

[0047] In step S22, the control section 12 uses the maximum path metric value, minimum path metric value and zero-state path metric value thus read out to calculate the S value in accordance with the expression (1) given hereinabove and stores the S value into the built-in memory thereof.

[0048] In step S23, the control section 12 compares the S value just determined by the calculation and a minimum value S_{min} of the S value calculated and stored in the past to discriminate whether or not the relationship of the two is $S < S_{min}$. If a result of the comparison is $S < S_{min}$, then the processing advances to step S24. It is to be noted that an initial value D of the minimum value S_{min} of the S value is a sufficiently high value. Accordingly, usually the S value calculated for the first time is discriminated to be lower than S_{min} ($= D$).

[0049] In step S24, the control section 12 discriminates whether or not the flag supplied from the CRC circuit 17 is 1 which indicates that no error is detected.

[0050] Here, processing of the CRC circuit 17 is described with reference to a flow chart of Fig. 8. It is to be noted

that the CRC process is executed in parallel to the EBP process of Fig. 7.

[0051] In step S31, the CRC circuit 17 extracts the zero-state path memory data stored in the EBP memory 16. In the present case, the CRC circuit 17 reads out the zero-state path memory data of the receive data E1 stored in the EBP memory 16.

[0052] In step S32, the CRC circuit 17 executes CRC calculation based on the zero-state path memory data of the receive data E1. In step S33, the CRC circuit 17 compares the CRC value obtained by the CRC calculation with data included in the zero-state path memory data and corresponding to the CRC value to discriminate whether or not the two data coincide with each other (whether or not the receive data includes an error). If the receive data includes no error, then the CRC circuit 17 supplies a CRC calculation result of the flag = 1 to the control section 12 (notifies the control section 12 that the receive data does not include an error) in step S34. If the receive data includes an error in step S33, then the CRC circuit 17 supplies a CRC calculation result of the flag = 0 to the control section 12 (notifies the control section 12 that the receive data includes an error).

[0053] Referring back to Fig. 7, if the flag representing the CRC calculation result is 1 in step S24, that is, if the receive data does not include an error, then in step S25, the control section 12 replaces the minimum value S_{min} of the S value in the built-in memory thereof with the S value just obtained by the calculation to store the S value as the minimum value S_{min} into the built-in memory. Further, the control section 12 discriminates that the length of the received data from the top of the receive data to the receive data E1 is the data rate and sets the data rate counter value to E1, and then stores the counter value into the built-in memory thereof.

[0054] On the other hand, if it is discriminated in step S23 that the comparison result is not $S < S_{min}$, then since the S value just calculated is not lower than the minimum value S_{min} of the S value, the data of the end bit of the receive data at the point of time, that is, in the present case, the receive data E1, is discriminated not to be EBP data. Thus, the processing advances to step S11 of the flow chart of Fig. 5.

[0055] Further, if it is discriminated in step S24 that the CRC flag inputted from the CRC circuit 17 is 0, then the control section 12 determines that the receive data includes an error. Thus, the processing advances to step S11 of the flow chart of Fig. 5.

[0056] Referring back to Fig. 5, in step S11, the control section 12 discriminates whether or not the receive data E1 just processed is the last one (last EBP) of receive data which may possibly be EBP data. If it is discriminated that the receive data E1 is the last one of such receive data, then the processing is ended. However, if it is discriminated that the receive data E1 is not the last one of receive data, then the processing returns to step S6.

[0057] For example, if the counter value illustrated in Fig. 6A is E2-1, that is, if the ACS circuit 14 tries to Viterbi decode the receive data E2-1 to E2+1 in one step, then it is discriminated in step S7 that the receive data E2+1 is not receive data which may possibly be EBP data. In other words, since the receive data E2 which may possibly be EBP data is not the last data to be Viterbi decoded in one step by the ACS circuit 14, the processing advances to step S12.

[0058] In step S12, the control section 12 executes an EBP data extraction process.

[0059] Here, the EBP data extraction process is described with reference to a flow chart of Fig. 9. In step S41, the control section 12 inhibits a write process of the data outputted from the ACS circuit 14 into the path metric path memory 15. In step S42, the control section 12 controls the receive data control section 13 to mask any one of the receive data E2-1 to E2+1 inputted from the receive data memory 11 which follows the receive data E1 which may possibly be EBP data and outputs resulting receive data to the ACS circuit 14. In particular, in the present case, the receive data E2+1 which follows the received data E2 is masked (the value of whose bit is set to 0), and the receive data E2-1 to E2+1 with the receive data E2+1 masked in this manner are outputted to the ACS circuit 14.

[0060] In step S43, the ACS circuit 14 Viterbi decodes the receive data E2-1 to E2+1 with the receive data E2+1 masked inputted thereto from the receive data control section 13 using data Viterbi decoded in the past and stored in the path metric path memory 15. Then, the ACS circuit 14 outputs a result of the Viterbi decoding to the path metric path memory 15 and outputs the maximum path metric value, minimum path metric value and zero-state path metric value of the receive data E2 to the EBP memory 16. At this time, since the path metric path memory 15 is in a write-inhibited state, the Viterbi decoding result of the ACS circuit 14 is not stored into the path metric path memory 15.

[0061] In step S44, the control section 12 executes an EBP process. The EBP process is similar to the process described hereinabove with reference to the flow chart of Fig. 7, and therefore, overlapping description of it is omitted herein to avoid redundancy.

[0062] In step S45, the control section 12 cancels the write-inhibited state of the path metric path memory 15.

[0063] Referring back to Fig. 5, in step S13, the control section 12 discriminates whether or not the receive data E2 just processed is the last receive data (last EBP data) which may possibly be EBP data. If the control section 12 discriminates that the receive data E2 is such last receive data, then the processing is ended. However, if the control section 12 discriminates that the receive data is not such last receive data, then the processing returns to step S4.

[0064] In this instance, in step S4, the ACS circuit 14 Viterbi decodes the receive data E2-1 to E2+1 (including the receive data E2+1 which is not masked) inputted thereto from the receive data control section 13 making use of data Viterbi decoded in the past and stored in the path metric path memory 15. Then, the ACS circuit 14 outputs a result of

the Viterbi decoding to the path metric path memory 15.

[0065] In other words, if receive data which may possibly be EBP data is not present at the end of receive data which are Viterbi decoded in one step by the ACS circuit 14 (if data which may possibly be EBP data is present at any other position than the end of the receive data), then the EBP data extraction process, that is, in the present example, a process denoted by (E2-1)' in Fig. 6B, is performed in step S12, and those data preceding to the receive data which is discriminated as EBP data (in the present example, the receive data E2-1 and E2) are Viterbi decoded, whereafter Viterbi-decoded data for the EBP (in the present case, Viterbi-decoded data of the receive data E2) is determined and stored into the EBP memory 16.

[0066] Thereafter, in step S4, a process of Viterbi decoding all receive data corresponding to the counter value E2-1, that is, Viterbi decoding the receive data E2-1 to E2+1 (a process denoted by E2-1 in Fig. 6B), is executed. In short, if data which may possibly be EBP data is preset at any other position than the end of those receive data which are to be Viterbi decoded in one step, then a Viterbi decoding process based on receive data corresponding to the same counter value is performed repetitively twice.

[0067] In this instance, the reason why writing into the path metric path memory 15 in the first Viterbi decoding process (EBP data extraction process) is inhibited is such as follows. In particular, since the ACS circuit 14 processes receive data inputted thereto from the receive data control section 13 using data Viterbi decoded in the past and stored in the path metric path memory 15, if a Viterbi decoding result based on data obtained by masking that or those of the receive data which is or are present following the receive data which are discriminated as EBP data is used for a Viterbi decoding process in a next step as in the process in step S42 of the flow chart of Fig. 9 (for example, if the data obtained by Viterbi decoding with the receive data E2+1 masked (set to 0) as described above is used for next Viterbi decoding), then an incomplete Viterbi decoding result is used in the Viterbi decoding, and this gives rise to an error. Therefore, in order to obtain a calculation result necessary for a next Viterbi decoding process, the second Viterbi decoding process is executed using the receive data which have not been masked, and a Viterbi decoding result of the second Viterbi decoding process is used in later Viterbi decoding processes.

[0068] In the foregoing description, the number of receive data (number of input data) which can be Viterbi decoded in one step by the ACS circuit 14 is 3 bits and the number of data of a result of the decoding is 1 bit. However, the number of receive data and the number of bits of a decoding result may be different bit numbers. For example, the number of receive data may be 4 bits and the number of a decoding result may be 2 bits.

[0069] The data demodulation apparatus described above can execute Viterbi decoding processing at a high speed with a simple circuit configuration and therefore can discriminate the data rate of receive data at a high speed.

[0070] While a preferred embodiment of the present invention has been described using specific terms, such description is for illustrative purposes only, and it is to be understood that changes and variations may be made without departing from the scope of the following claims.

[0071] In so far as the embodiments of the invention described above are implemented, at least in part, using software-controlled data processing apparatus, it will be appreciated that a computer program providing such software control and a storage medium by which such a computer program is stored are envisaged as aspects of the present invention.

Claims

1. A data demodulation apparatus for receiving data of an arbitrary one of a plurality of data rates and demodulating the receive data making use of blind transport format detection, comprising:

Viterbi decoding means for successively Viterbi decoding the receive data beginning with the top bit; end bit position discrimination data extraction means for successively extracting end bit position discrimination data which are obtained at end bit positions of the data rates when the receive data are successively Viterbi decoded beginning with the top bit by said Viterbi decoding means; and data rate discrimination means for discriminating the data rate of the receive data based on the end bit position discrimination data successively extracted by said end bit position discrimination data extraction means.

2. A data demodulation apparatus according to claim 1, further comprising end bit position discrimination data storage means for storing the end bit position discrimination data successively extracted by said end bit position discrimination data extraction means.

3. A data demodulation apparatus according to claim 1, wherein said Viterbi decoding means Viterbi decodes a plurality of bits of the receive data in one step.

4. A data demodulation apparatus according to claim 3, wherein, if a bit of the receive data which may possibly be an end bit position bit is not the last bit of the plurality of bits which are Viterbi decoded in one step by said Viterbi decoding means, then said Viterbi decoding means first Viterbi decodes those of the receive data to be Viterbi decoded in one step from the top of the receive data to the bit which may possibly be an end bit position bit, and then Viterbi decodes the remaining one or ones of the receive data up to the last bit of the plurality of bits.
5. A data demodulation apparatus according to claim 4, further comprising Viterbi-decoded data storage means for storing data Viterbi decoded by said Viterbi decoding means, and wherein, if the bit of the receive data which may possibly be an end bit position bit is not the last bit of the plurality of bits which are Viterbi decoded in one step by said Viterbi decoding means, then when said Viterbi decoding means Viterbi decodes those of the receive data to be Viterbi decoded in one step from the top of the receive data to the bit which may possibly be an end bit position bit, said Viterbi-decoded data storage means is inhibited from storing the Viterbi decoded receive data.
6. A data demodulation apparatus according to claim 1, wherein the end bit position discrimination data include a maximum path metric value, a minimum path metric value, a zero-state path metric value and zero-state path memory data.
7. A data demodulation method for a data demodulation apparatus which receives data of an arbitrary one of a plurality of data rates and demodulates the receive data making use of blind transport format detection, comprising:
 - a Viterbi decoding step of successively Viterbi decoding the receive data beginning with the top bit;
 - an end bit position discrimination data extraction step of successively extracting end bit position discrimination data which are obtained at end bit positions of the data rates when the receive data are successively Viterbi decoded beginning with the top bit by the process of the Viterbi decoding step; and
 - a data rate discrimination step of discriminating the data rate of the receive data based on the end bit position discrimination data successively extracted by the process of the end bit position discrimination data extraction step.

FIG. 1

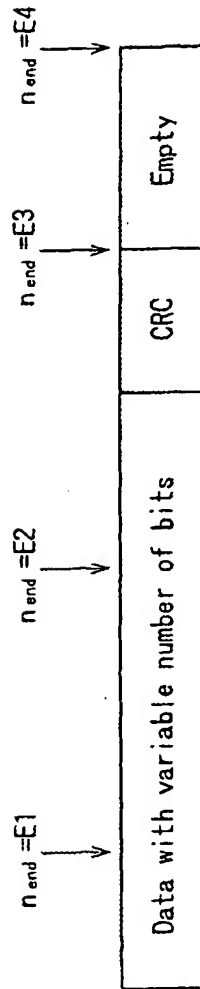


FIG. 2

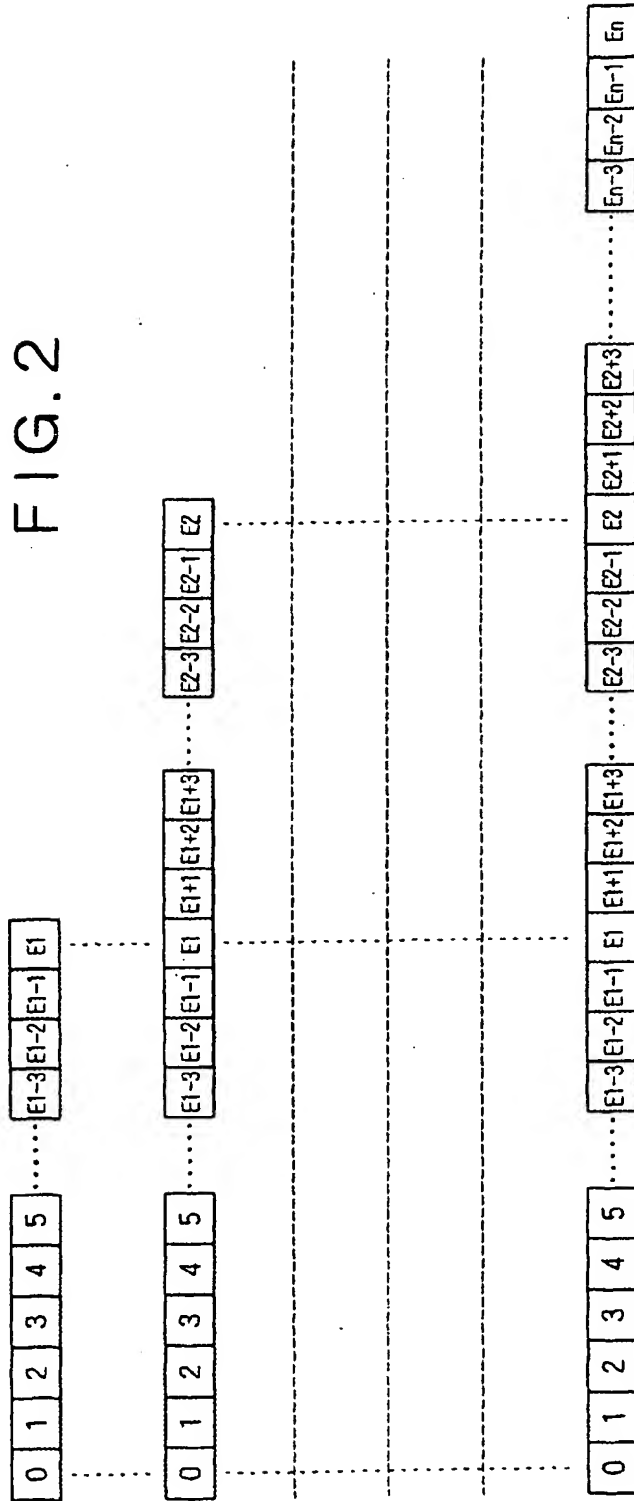


FIG. 3

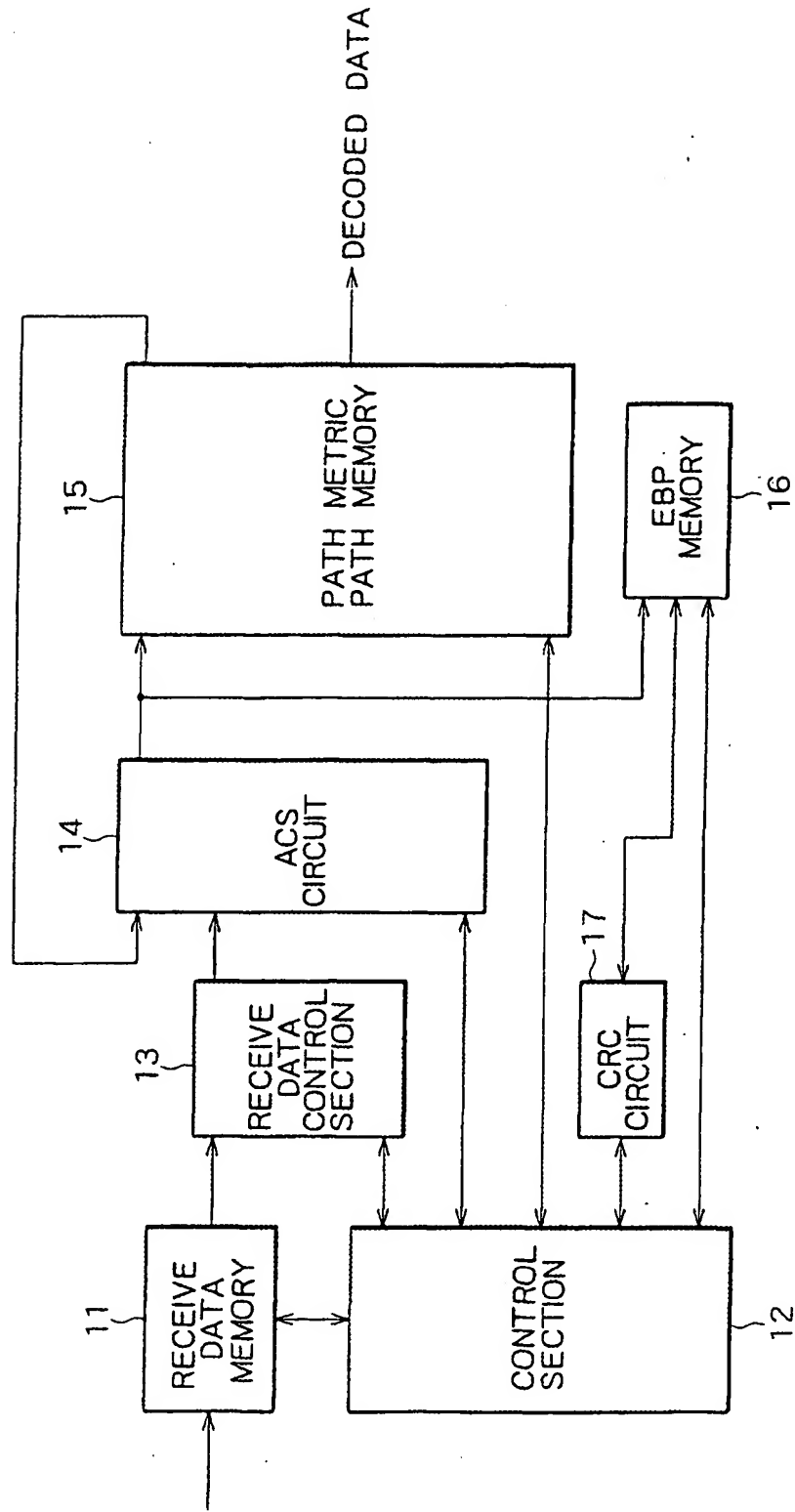
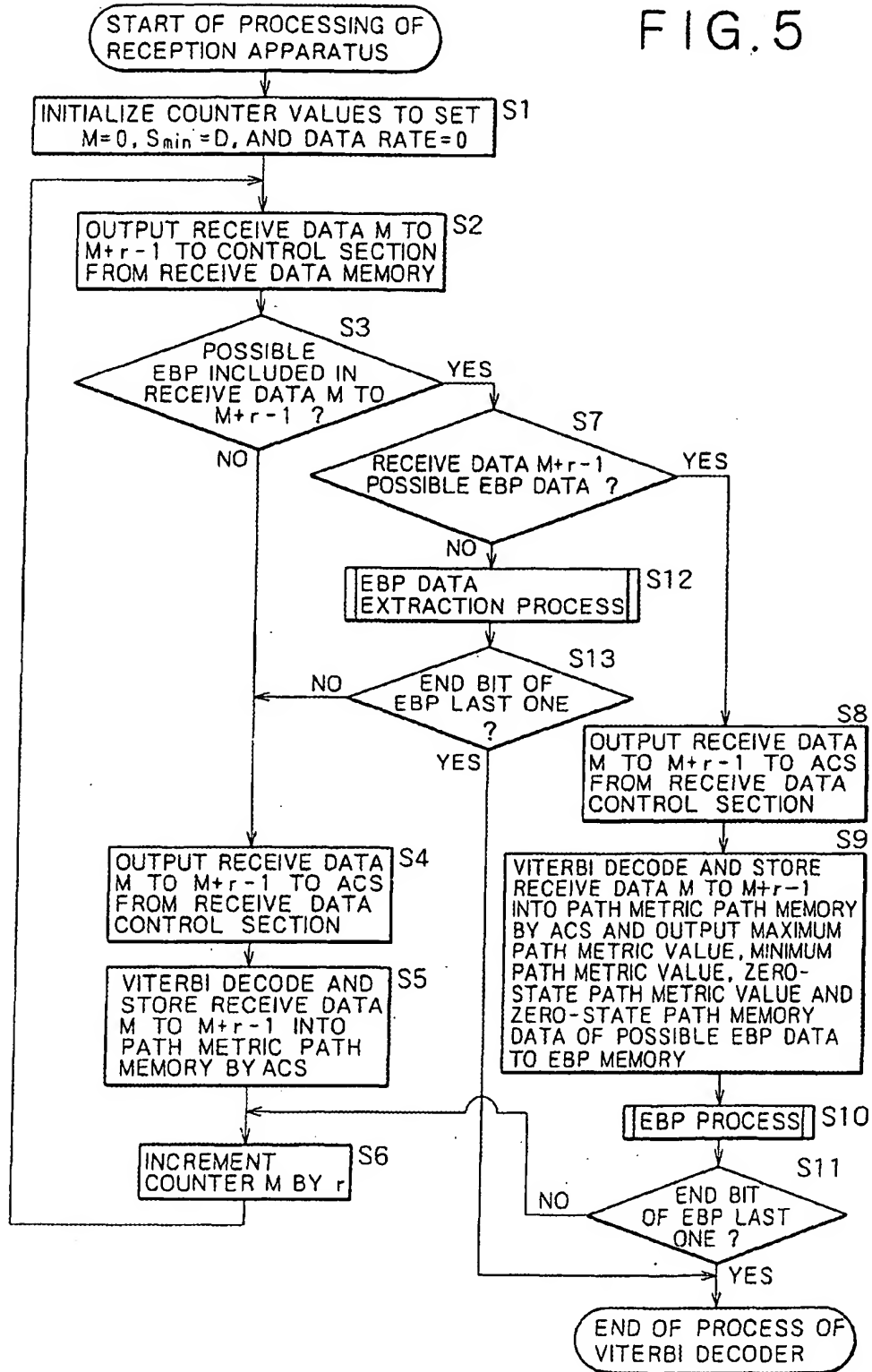


FIG. 4



FIG. 5



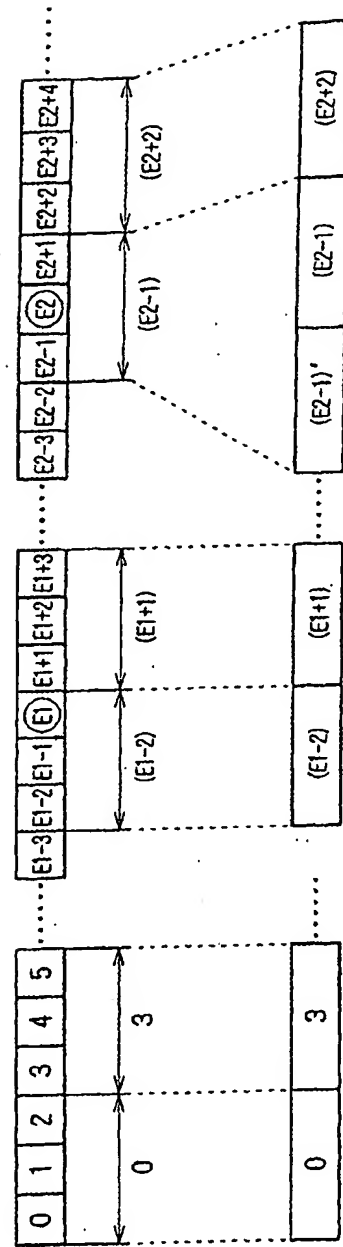


FIG. 6A

FIG. 6B

FIG. 7

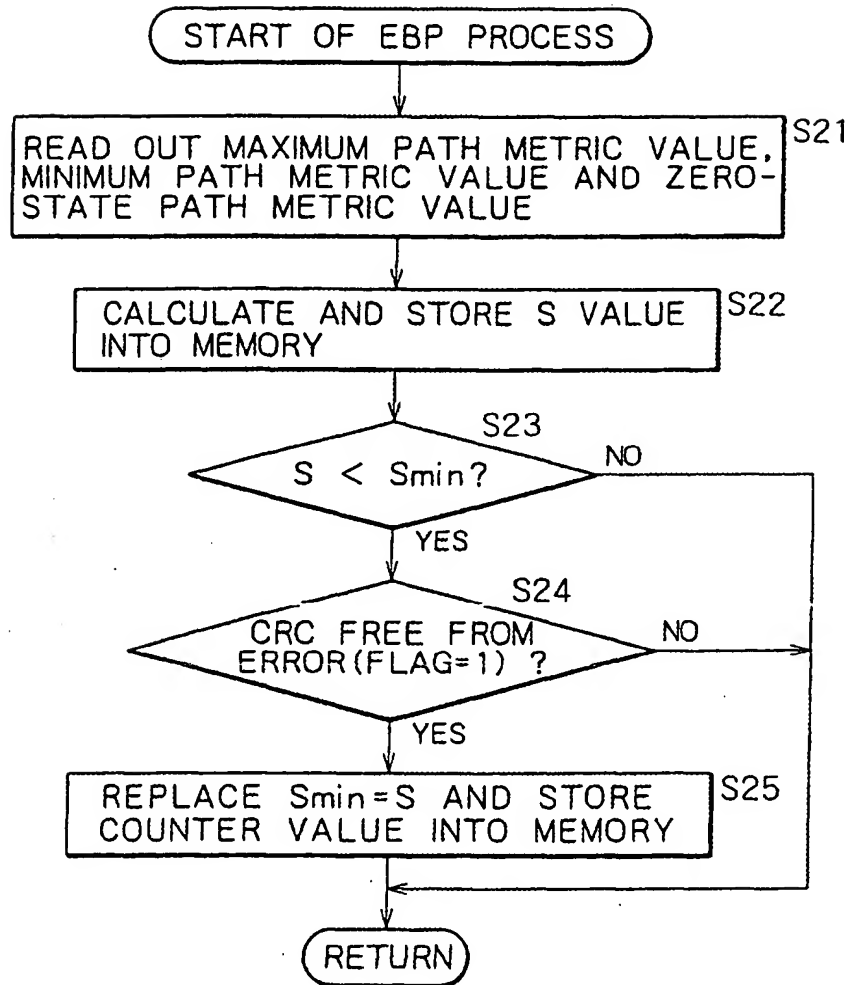


FIG. 8

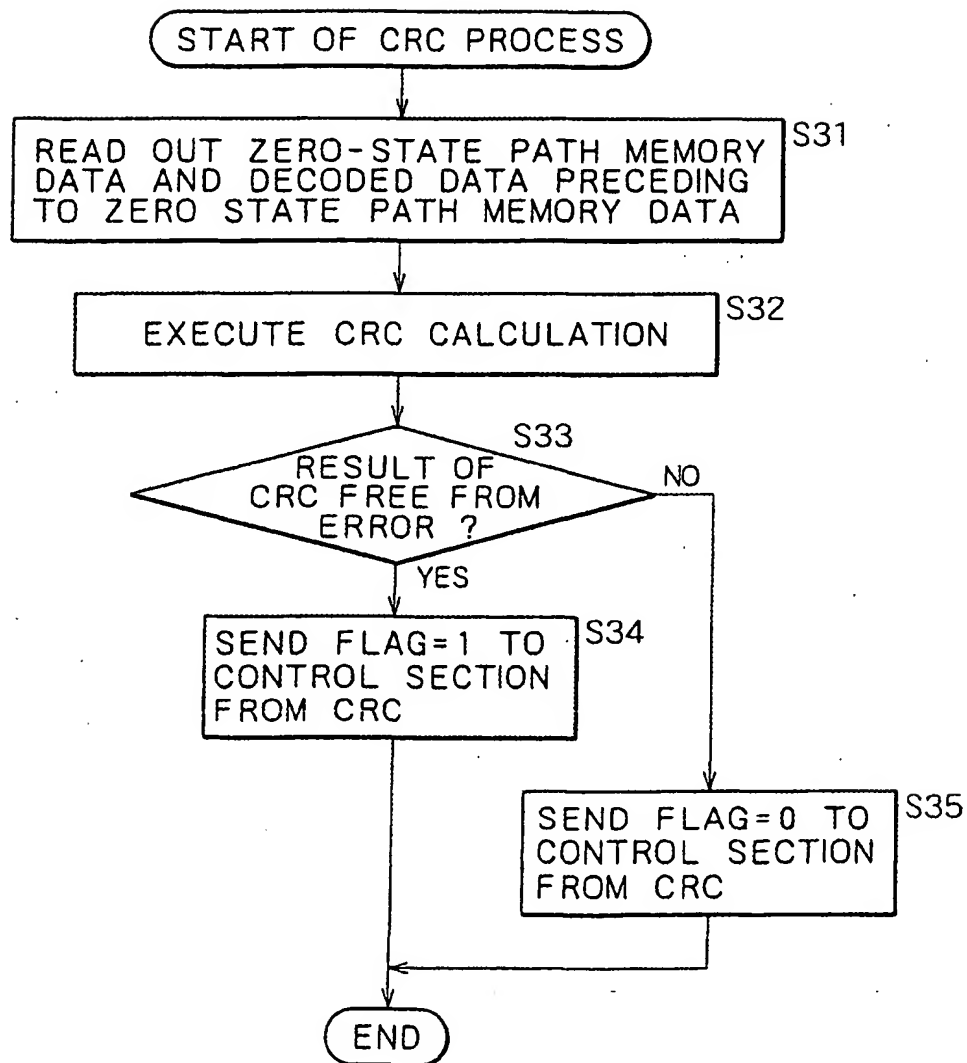


FIG. 9

